KEMI-TORNIO UNIVERSITY OF APPLIED SCIENCES TECHNOLOGY

Li Xu

Symbian S60 Supermarket Administration System

The Bachelor's Thesis Technology programme
Kemi 2010

PREFACE

This Bachelor's Thesis is an individual project mainly for the self-improving through the whole project development step by step at the Department of information Technology.

I would like to thank my supervisors Aalto Teppo and Pekka Harjuniemi, who are the teachers in Kemi-Tornio University of Applied Science. I appreciate your technical advice and understanding.

Kemi, 2 Sep 2010
Li Xu

ABSTRACT

Kemi-Tornio University of Applied Sciences, Technology
Degree Programme Electrical Engineering
Name Li Xu
Title Symbian supermarket administration system
Type of Study Bachelor's Thesis
Date 17 May 2010
Pages
Instructor Aalto Teppo and Pekka Harjuniemi
Company
ContactPerson/Supervisor Aalto Teppo and Pekka Harjuniemi
From Company

The goal of this project is to design a Symbian S60 system that display short messages and administrate the goods that on sale in the supermarket by using S60 cell phone. Since the mobile phone is developing very fast, more and more cell phone is becoming like a mini computer beside it can make or receive a call. NOKIA are using Symbian Operating System to build a new cell phone management networks, so why and how to construct this system become a more and more important for IT engineers. While programming on this system, QT is the latest software than Carbide C++ for Symbian S60.

Keywords: Symbian S60, QT, Administration system.

CONTENTS

EXPLANATION OF CHARCTERS AND ABBREVIATIONS

3G	Third Generation
GPRS	General Packet Radio service
GSM	Global System for Mobile communications
S60	Symbian S60 Platform
Qt	Qt programming Software
TCP/IP	Transmission Control Protocol/ Internet Protocol
UML	Unified Modelling Language

1. INTRODUCTION

The Symbian S60 system is becoming more and more common Mobile System of the cell phone which produced by NOKIA. The purpose of this thesis is to study and build a kind of software based on the S60 System, that main task is on supermarket items administration, and also can let the customer can find the resources buy themselves in this region.

The basic structure of this system could not only be used in supermarket but also can be used in enterprises and organizations. The customer and the administrator will connect the device administrator (electron) by GPRS or 3G networks. The features of the GPRS and 3G net works will be introduced in chapter 2. This chapter will also sore out the features of Carbide C++ and QT programming software.

The object of this project is about to design a system based on the Symbian S60 platform by using Carbide C++ and QT programming software. It can let the people to choose the goods which they want to buy by using their mobile devices. This system could also be involved in many other fields like industrial management, business and so on. The features of Symbian S60 platform will be sore out in chapter 3.

The basic structure of this system will be map out in chapter 4, I have learned both about tools and design, get familiar with Symbian S60 system, using the Carbide C++ and QT programming software as a tool to design the human interface of this system will also been told in chapter 4.

And in Chapter 5, the system can be simulated by a Symbian system installed cell phone to test its reliability. The testing plane and the expecting result will be soured out in this chapter.

At the end of this project in Chapter 6, I will make a conclusion about its Advantage and Disadvantage according to the resources been sore out in this thesis.

2. SOFTWARE AND COMMUNICATION

The description about the software and communication ways in this project presents a quick view of major process and key process participants. The description also includes the relationships between each interface which is involved into this system.

2.1. GPRS

General packet radio service (GPRS) is a packet oriented mobile data service available to all users of the 2G cellular communication systems global system for mobile communications (GSM), as well as in the 3G systems. In 2G systems, GPRS provides data rates of 56-114 Kbit/second. /1/

2.1.1. Introduction

2G cellular technology combined with GPRS is sometimes described as 2.5G, that is, a technology between the second (2G) and third (3G) generations of mobile telephony. It provides moderate-speed data transfer, by using unused time division multiple access (TDMA) channels in, for example, the GSM system. GPRS is integrated into GSM Release 97 and newer releases. It was originally standardized by European Telecommunications Standards Institute (ETSI), but now by the 3rd Generation Partnership Project (3GPP)./1/

GPRS usage charging is based on volume of data, either as part of a bundle or on a pay as you use basis. An example of a bundle is up to 5 GB per month for a fixed fee. Usage above the bundle cap is either charged for per megabyte or disallowed. The pay as you use charging is typically per megabyte of traffic. This contrasts with circuit switching data, which is typically billed per minute of connection time, regardless of whether or not the user transfers data during that period./1/

GPRS is a best-effort packet switched service, as opposed to circuit switching, where a certain quality of service (QoS) is guaranteed during the connection. GPRS was developed as a GSM response to the earlier CDPD and i-mode packet switched cellular technologies./1/

2.1.2. Usability

The maximum speed of a GPRS connection offered in 2003 was similar to a modem connection in an analogue wire telephone network, about 32-40 kbit/s, depending on the phone used. Latency is very high; round-trip time (RTT) is typically about 600-700 ms and often reaches 1s. GPRS is typically prioritized lower than speech, and thus the quality of connection varies greatly./1/

Devices with latency/RTT improvements (via, for example, the extended UL TBF mode feature) are generally available. Also, network upgrades of features are available with certain operators. With these enhancements the active round-trip time can be reduced, resulting in significant increase in application-level throughput speeds./1/

2.2. 3G

"Third generation" redirects here. For third-generation immigrants./2/

International Mobile Telecommunications-2000 (IMT--2000), better known as 3G or 3rd Generation, is a generation of standards for mobile phones and mobile telecommunications services fulfilling specifications by the International Telecommunication Union./2/

Application services include wide-area wireless voice telephone, mobile Internet access, video calls and mobile TV, all in a mobile environment. Compared to the older 2G and 2.5G standards, a 3G system must allow simultaneous use of speech and data services, and provide peak data rates at least 200 Kbit/s according to the IMT-2000 specification. Recent 3G releases, often denoted 3.5G and 3.75G, also provide mobile broadband access of several Mbit/s to laptop computers and smartphones./2/

2.2.1. Data Rates

ITU has not provided a clear definition of the data rate users can expect from 3G equipment or providers. Thus users sold 3G service may not be able to point to a standard and say that the rates it specifies are not being met. While stating in commentary that "it is expected that IMT-2000 will provide higher transmission rates: a minimum data rate of 2 Mbit/s for stationary or walking users, and 384 Kbit/s in a moving vehicle," the ITU does not actually clearly specify minimum or average rates or what modes of the interfaces qualify as 3G, so various rates are sold as 3G intended to meet customers expectations of broadband data./2/

2.2.2. Security

3G networks offer greater security than their 2G predecessors. By allowing the UE (User Equipment) to authenticate the network it is attaching to, the user can be sure the network is the intended one and not an impersonator. 3G networks use the KASUMI block crypto instead of the older A5/1 stream cipher. However, a number of serious weaknesses in the KASUMI cipher have been identified./2/

In addition to the 3G network infrastructure security, end-to-end security is offered when application frameworks such as IMS are accessed, although this is not strictly a 3G property./2/

2.3. GSM

GSM (Global System for Mobile Communications: originally from Group Special Mobile) is the most popular standard for mobile telephony systems in the world. The GSM Association, its promoting industry trade organization of mobile phone carriers and manufacturers, estimates that 80% of the global mobile market uses the standard. GSM is used by over 1.5 billion people across more than 212 countries and territories. Its ubiquity enables international roaming arrangements between mobile phone operators, providing subscribers the use of their phones in many parts of the world. GSM differs from its predecessor technologies in that both signalling and speech channels are digital, and thus GSM is considered a second generation (2G) mobile phone system. This also facilitates the wide-spread implementation of data communication applications into the system./3/

2.3.1. Network Structure

The network is structured into a number of discrete sections; it will be shown as Figure 1:

- Includes all the functionalities of normal mobile phone, such as message sending, and so on./3/
- Wireless connection, which supports GSM network (GPRS), or 3G network./3/
- The GPRS Core Network (the optional part which allows packet based Internet connections)./3/
- The Operations support system (OSS) for maintenance of the network./3/

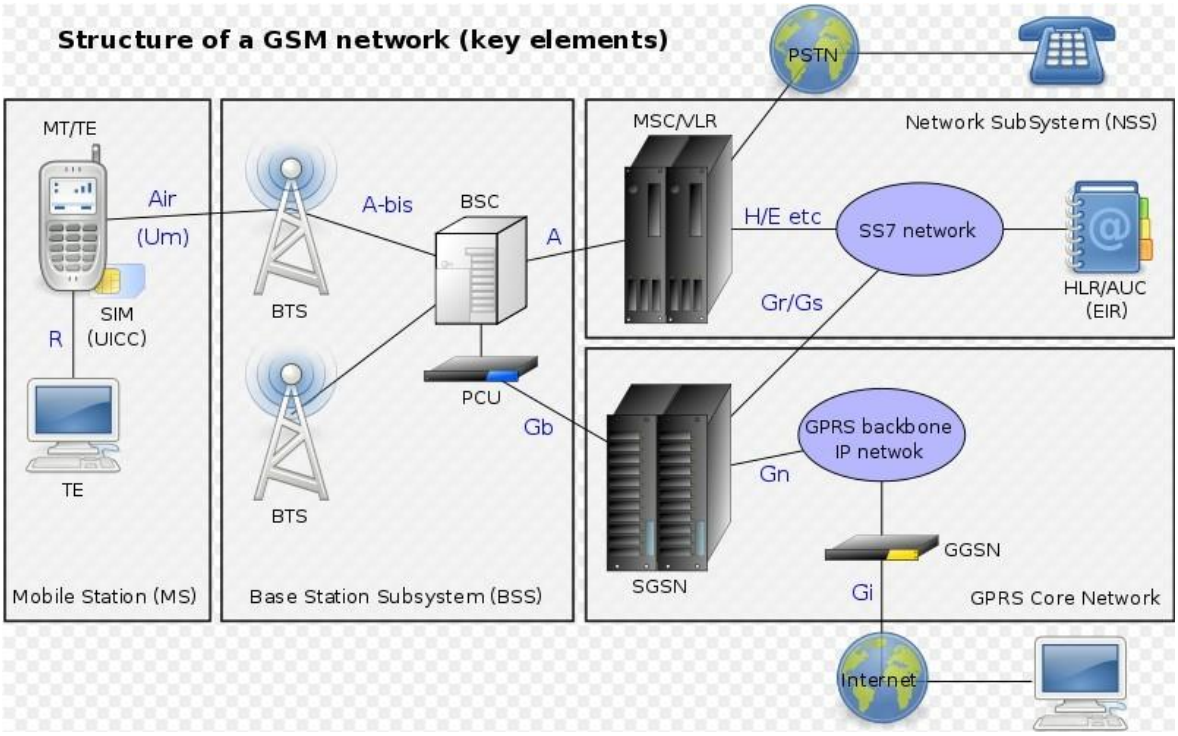

Figure.1 Structure of The GSM Net work/3/

2.3.2. Service Security

GSM was designed with a moderate level of service security. The system was designed to authenticate the subscriber using a pre-shared key and challenge-response. Communications between the subscriber and the base station can be encrypted. The development of UMTS introduces an optional Universal Subscriber Identity Module (USIM), that uses a longer authentication key to give greater security, as well as mutually authenticating the network and the user - whereas GSM only authenticates the user to the network (and not vice versa). The security model therefore offers confidentiality and authentication, but limited authorization capabilities, and no non-repudiation./3/

2.4. Software

In the following steps, I will use both Carbide C++ and QT for programing and Layout Design. Both of these software are now developed by Nokia and can be down load form Nokia forum. Figure 2 will show the interface of Carbide C++.

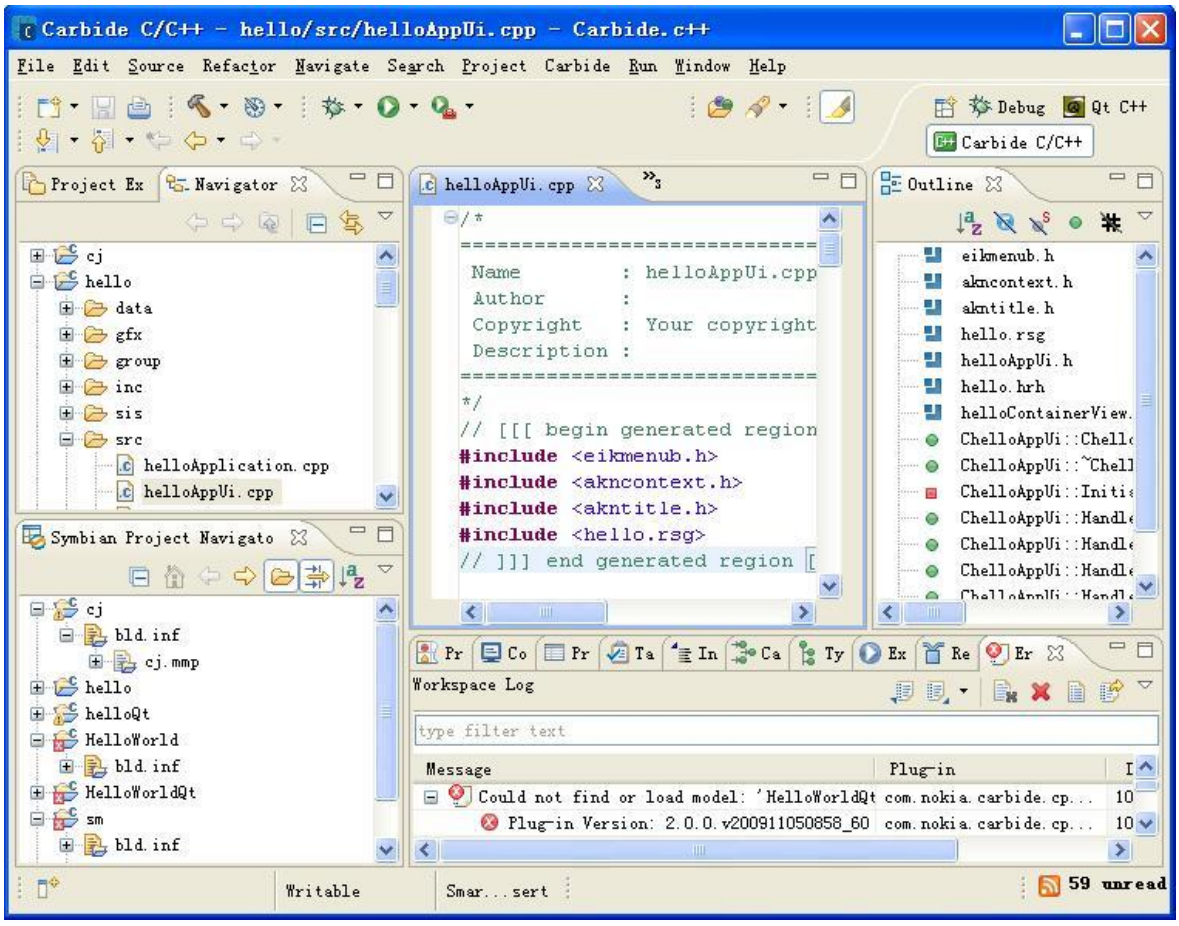

Figure 2. Carbide C++ V.2.3.0

The Carbide C++ IDE is designed from the ground up for developers creating Symbian C++ and Qt software for Symbian devices. It provides features for project creation, drag-

and-drop UI design, querying API information, investigating project dependencies, performing static code analysis, building and packaging software, undertaking emulator and on-device debugging, and investigating project performance. /8/

In this project, the Carbide C++ will be used for the Application interface design and also for the future coding and decoding and also code debugging, because I am familiar with the Carbide C++, so I prefer to use this software to start this project.

Qt is a cross-platform application and UI framework. The designer can use Qt to write web-enabled applications once and deploy them across desktop, mobile and embedded operating systems without rewriting the source code. The modular Qt C++ class library provides a rich set of application building blocks, delivering all of the functionality needed to build advanced, cross-platform applications. The platforms for Qt are: Embedded Linux, Mac OSX, Windows, Linux/X11, Windows Mobile, Windows CE, Symbian, Maemo and MeeGo./9/

Figure 3 is the interface of Qt. This software been used in this project is to support Caribide C++ as an advanced emulator and debugger.

Figure.3 Qt Creator 1.3.1

Qt Creator is a cross-platform integrated development environment (IDE) tailored to the needs of Qt developers. It provides:/9/

- C++ and JavaScript code editor/9/

- Integrated UI designer /9/
- Project and build management tools/9/
- gdb and CDB debuggers/9/
- Support for version control/9/
- Simulator for mobile Uis/9/
- Support for desktop and mobile targets/9/

3. SYMBIAN S60 OPERATING SYSTEM

Symbian OS is a proprietary operating system designed for mobile devices, with associated libraries, user interface, frameworks and reference implementations of common tools, developed by Symbian Ltd. It is a descendant of Psion's EPOC and runs exclusively on ARM processors although a non-productized x86 port exists. The S60 Platform (formerly Series 60 User Interface) is a software platform for mobile phones that runs on Symbian OS. /7/

3.1. Introduction

Symbian OS is one of Nokia's mobile operating systems for mobile devices and smart phones, with associated libraries, user interface, frameworks and reference implementations of common tools, originally developed by Symbian Ltd. It was a descendant of Psion's EPOC and runs exclusively on ARM processors, although an unreleased x86 port existed./4/

Symbian OS and its associated user interfaces S60, UIQ and MOAP(S) were contributed by their owners to the foundation with the objective of creating the Symbian platform as a royalty-free, open source software./4/

Till now S60 platform has been divide into different kind of series, from the earliest 1st edition to the latest 5th edition.

3.2. Symbian S60 SDK

The Symbian SDKs could enable users to build applications using Qt, Symbian C++, Java™ technology, and Web Runtime (WRT). In addition, several of the SDKs support development with Python for S60 using an SDK plug-in. When used with the users' personal favourite development tools, the SDKs provide them with access to all the APIs implemented in a particular version of Symbian or edition and feature pack of Nokia S60 on Symbian OS. Once they've coded their application, the SDKs include tools to compile that application for testing in the SDK's emulator. In the emulator, the application can be tested without the need for a Symbian device, which improves development time. Once the application is tested, tools to compile and package the application for installation on a Symbian device are provided. /9/

The SDKs include comprehensive documentation and example applications that enable you to make the most of Symbian and the SDK tools./9/

3.3. Features

Based on EPOC 32 OS, which was released at July 1997, the first Symbian OS came out at June called Symbian 6.0, although the manufacturers have some part of its source code, but it is not an open source software, SIS files which can also be easily transferred by using Bluetooth, or through the internet or through transfer using cables. Symbian S60 platform is currently amongst the most-used smart phone platforms in the world; include the 1st-3rd edition for keyboard smart phone and 5th edition for touch screen smart phone.

3.3.1. Main Components

Figure.4 is showing the steps of how the Symbian application will be installed in the devices driver/hardware. These steps also contain the main components which need to create a Symbian system.

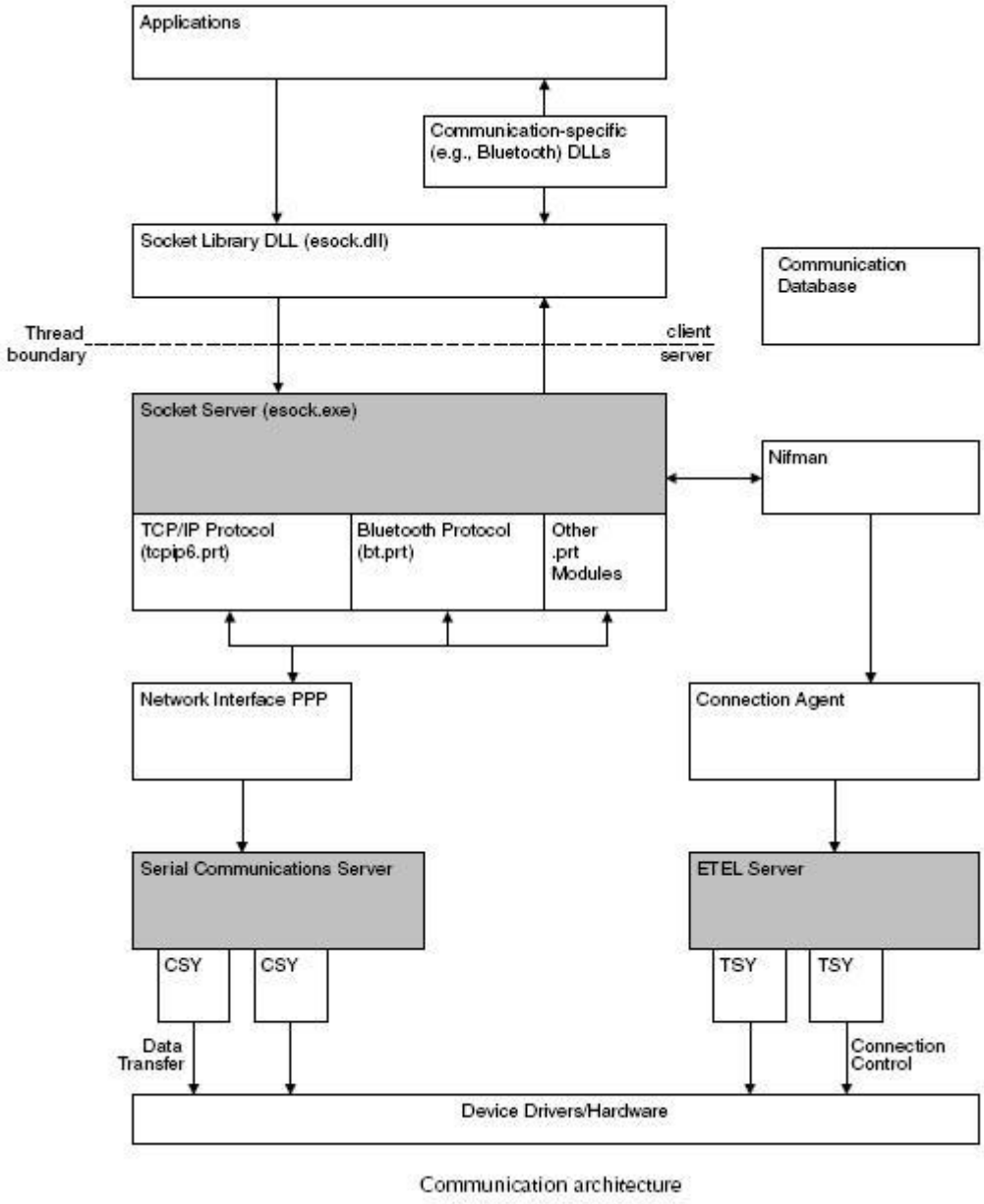

Figure.4 Components of the Symbian communication architecture/5/

3.3.1.1. Applications and DLLs

Applications use networking API classes in DLLs to access communications features. As with other application-level DLLs, the communication DLLs hides the details of the underlying architecture. Symbian OS provides a socket-based API that operates in a similar way to the BSD socket API./5/

3.3.1.2. Communication-Specific Functionality

In addition to the socket API, there are also APIs specific to certain types of communication, such as Bluetooth. A Bluetooth program would call functions from the Bluetooth DLL for device discovery, for example, and then use the socket API for the bulk of the data communications./5/

3.3.1.3. Socket Server

The socket server is a process that implements and manages communication sockets. Applications act as clients to this server through the application-level communication DLLs. As with the GUI DLLs, these functions hide the actual client/server communications from the socket server./5/

3.3.1.4. Protocol Modules

The socket server uses protocol modules for handling the network data protocols. These are polymorphic DLLs (prt files) that implement different communication protocols, while providing a consistent interface to the socket server. Examples of protocol modules are TCP/IP, Bluetooth and IR. New protocol modules can be created and used. Protocol modules are independent of the data-link layer – bringing up the connection and exchanging data with the device is done through an abstracted interface. This interface is accomplished with two other plug-in modules that attach to the socket server: a network interface (which is usually the PPP module) and a connection agent. /5/

3.3.1.5. Network Interface Manager(Nifman)

The socket server with protocol modules use Nifman (network interface manager) to establish the connection and set up the data path to the data link level. In order to start a particular physical connection, Nifman will load a connection agent. /5/

3.3.1.6. Connection Agent

A connection agent is a polymorphic DLL that is responsible for starting and stopping the communication connection. Not only is it responsible for establishing the connection itself (e.g. dialing a number for GSM or starting GPRS), but it provides information to set up the data communication path between the physical device and the network protocol module. The connection agent will normally use ETEL to start the connection./5/

3.3.1.7. Communication Database

A connection agent will consult the communication database to determine how to establish the network connection. This database contains all the settings applicable to communication connections. Depending on database settings, the connection agent can choose to start a preferred connection or it may prompt the user to select a connection.

Once the connection is chosen, the agent will extract all the applicable connection parameters from the database to start the connection. /5/

A connection to a network on a Symbian OS smartphone is known as an Internet Access Point (IAP). An example is a GSM CSD connection using a specific ISP phone number and login information – all stored in the communication database entries for that IAP. /5/

3.3.1.8. ETEL Server

ETEL is a low-level server used to establish a connection with a communication device. It provides an abstracted telephony API to its clients, with functions for tasks such as establishing the connection, terminating the connection, and retrieving line status and device capabilities. Modules, called TSY modules, are installed to contain the implementation for the target device. An ETEL client will load the appropriate TSY, then use the ETEL-abstracted API to control the device. Symbian OS has many built-in TSYs for devices such as GPRS and GSM (files end in .tsy). /5/

In addition to loading a Connection Agent, Nifman will load a network interface module (DLLs suffixed with .nif). This is usually PPP.nif which implements the PPP data link protocol. This module uses the abstracted API of the Communications Server to transfer data to the device./5/

3.3.1.9. Serical Communications Server

The Serial Communications Server provides an abstraction for serial communication across multiple devices. Reading and writing serial data and managing data flow control are example functions of this abstracted API. The details of the low-level protocols for handling a specific device are implemented in DLL CSY modules (suffixed by .csy). Example CSY modules include IR, GPRS and UART./5/

3.3.1.10. CSY Modules

CSY modules communicate with the hardware through device drivers. The device drivers handle the actual control of the communications hardware. Symbian OS v7.0 and previous versions could have only one active IAP connection at a time. Symbian OS v7.0s introduced a multi-homing capability: the ability to have multiple IAP connections – each with its own IP address – active at once. This is useful, for example, if you want multiple functions active that use different GPRS contexts (such as MMS and web browsing). Another example is having interfaces such as WLAN and GPRS up at the same time. /5/

This feature opens up many possibilities for devices that support multiple ways of accessing the Internet and will become more important for future smartphones. /5/

3.3.2. Types of Memory in Symbian OS

There are total four types of memory in Symbian OS. The different types of memory are been used in different places.

3.3.2.1. Radom Access Memory(RAM)

RAM is the volatile execution and data memory used by running programs. Applications vary in how much RAM they use, and this also depends on what the application is doing at the time. For example, a browser application loading a web page needs to allocate more RAM for the web page data as it's loaded. Also, the more RAM space you have, the more programs you can run on your smartphone at once. Typically, mobile phones have between 7 and 30MB of RAM available for applications to use./6/

3.3.2.2. Read Only Memory(ROM)

The ROM is where the Symbian OS software itself resides. It includes all the startup code to boot the device, as well as all device drivers and other hardware-specific code. This area cannot be written to by a user, although some of it can be seen by the file system as drive will show all the built-in applications of the OS, as well as the system DLLs, device drivers and system configuration files. For added efficiency, code in ROM is executed in place – i.e. it is not loaded into RAM before executing. Typically a phone has between 16 and 32MB of ROM./6/

3.3.2.3. Internal Flash Disk

The internal flash acts like a disk drive and allows for reading and writing files to it via the Symbian OS file system. The file system is fully featured and supports a hierarchical directory structure, with very similar features to those you would find on high-end operating systems. The internal flash drive is represented as the c: drive to the file system. This memory contains user-loaded applications, as well as data such as documents, pictures, video, bookmarks, calendar entries, etc. /6/

The size of the internal flash disk varies with the phone, but it can be quite generous. For example, the Nokia 9500 has 80MB of internal flash space available to the user. On many phones, however, available internal user space is significantly less. The Nokia 6600, for example, has 6MB of flash space available to the user./6/

3.3.2.4. Removable Memory Cards

Memory cards act as removable disk drives and allow you to expand the storage provided internally. You can also read from and write to a memory card just as to the internal disk – including operations such as saving user data, and even installing applications. This card is treated as another disk volume by the file system and is represented by a drive letter such as d: or e: (this varies between phones). The memory card formats (MMC and SD are

examples) and available sizes vary by phone. Memory card sizes can vary from 16MB (or even less) to 1GB. /6/

4. DESIGN

In this project, I will use Qt and Carbide to programming an Administration System on the Symbian S60 platform. But at first I will sore out the basic structure of this system, so that to make a clear plan for the future process.

4.1. Introduction of Design

This Administration system based on Symbian S60 platform, and its functionalities are worked through Short Message Service communication and local database transmission. The system accomplishes Real-Time working principle and remotes control by smart-phone, It should be constructed as a secure, convenient, flexible, fast as an economical administration system.

Once the customer and the administrator have installed the application into their cell phone, the application will be connected with server through 3G or GPRS network any where. The application is designed dynamically, and it follows the GUI principle, which allows user to interact with application as simple as possible.

Figure.5 is showing how this kind of system works; it includes server, database, device administrator and security department. And they transmit the data between customer and device administration with their mobile device by using the GPRS and 3G net works. But between server and database and security department, they will use internet broadband to increase the speed of data transmission.

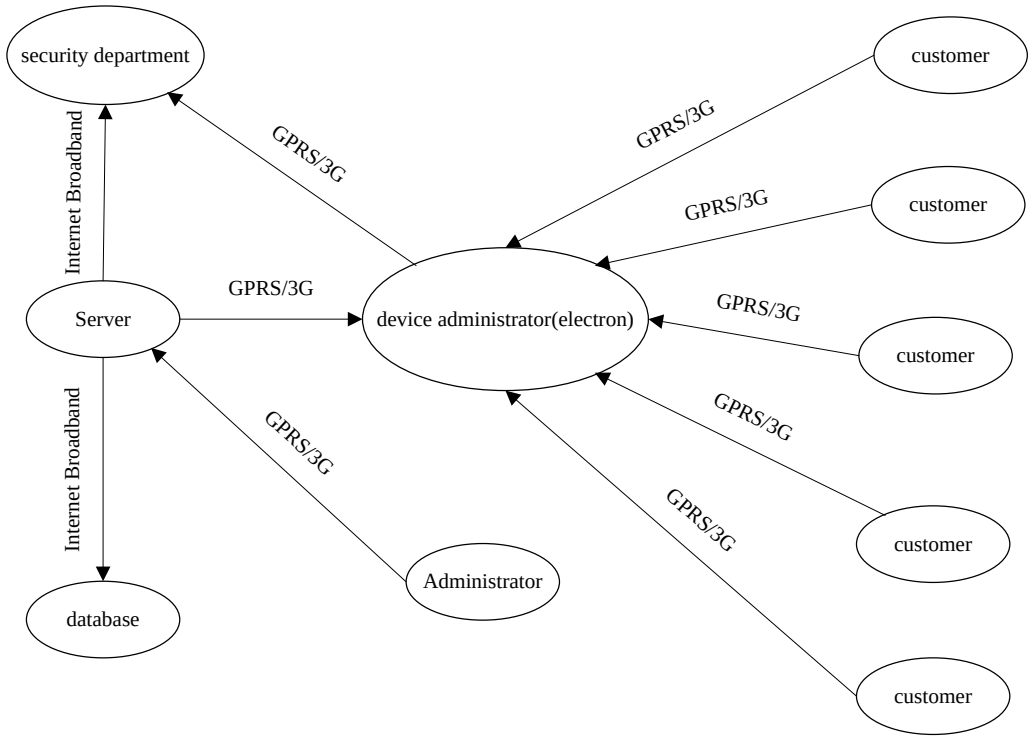

Figure.5 The Basic Structure

The first step of design this system is the structure of the interface design, the whole system development based on the structure of the interface design. After it has been completed and in order to accomplish a Real-Time controlling, it is necessary to access a database through out of the system to embed its application. But according to this project's schedule, there is not enough time to create such a database, so at the part of the designation, I will not discuss a lot about the database, so there is not too much information about it. This database installed the basic information about the customer and administrator's ID password, the information about the items on sell and so on.

In the programming part, I will use the Carbide C++ which installed with Qt as a tool of software development and the programming language I will use C++.

While the semaphore engine accomplishes the functionality, its engine is always keeping watch on the application when it has been launched, and it will operate with any modification from the UI or server at any time.

4.2. Connection

This system has been designed as a mobile system, so it will use GPRS or 3G network to be connected between the User/Administrator and the Server. The structure and the security ways are important for this system.

4.2.1. Connection Structure

If the customers and administrators want to read the data from the database through the system or access the data, they must have their own information so they must be registered. Everyone can update and change their own data, but only administrator can access the data of the items on sell, they have no authority to access other registers' data. And all this will be implemented with the GPRS or 3G net work.

A search engine should also been built up in the application. This functionality is defined to help the user to find the current principals who are taking charge in this studio at that moment. The user also can get the list of the principal's contact information, and take it into action. The in case that to guarantee the searching speed, the server and the database will use internet broadband to be connected between each other.

The extra way is the news/restock and discontinued searching engine in the application. This functionality is defined that user can find out the new principals or which back form stock for a long time and also some information long ago. It will let the user to find out which they expected, the principal of this extra searching engine is the same as the basic searching engine.

The ways of connection is important, it should to protect the data and create an envelope for each kind of user, either customer or administrator. Figure 6 is showing the structure of connection. The devices must be managed all the time, and the administrator should inform the principals immediately according the contact information.

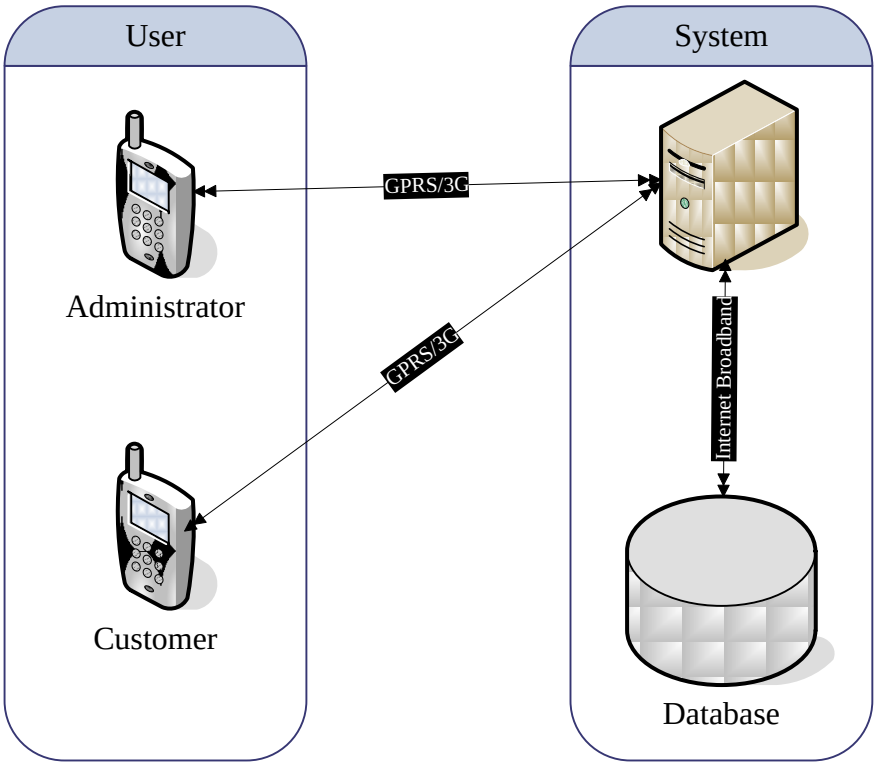

Figure 6. The Structure of Connection

4.2.2. Security

It will map out the secure functionalities of this administration software at the beginning of the designation. The secure functionalities were considered as the security of customers' and administrators' benefits. So these functionalities were designed according to the requirements from customers and also can let the administrator can manage the data fluently.

There are three main modules in this System:

- Trusted computing base.
- Data caging.
- Capabilities.

4.3. Database Module

This database is mapping out of the system to show the main data which is needed for the future design.

Here in Figure 7, the Facility contains all of the interfaces that been connected with Menu. It depends on the authority of the user who long in to the system. The Administrator has more authority than the customer as they can upload and delete the data from database. The definition of all the keys in these tables will be seen in Appendix 1.

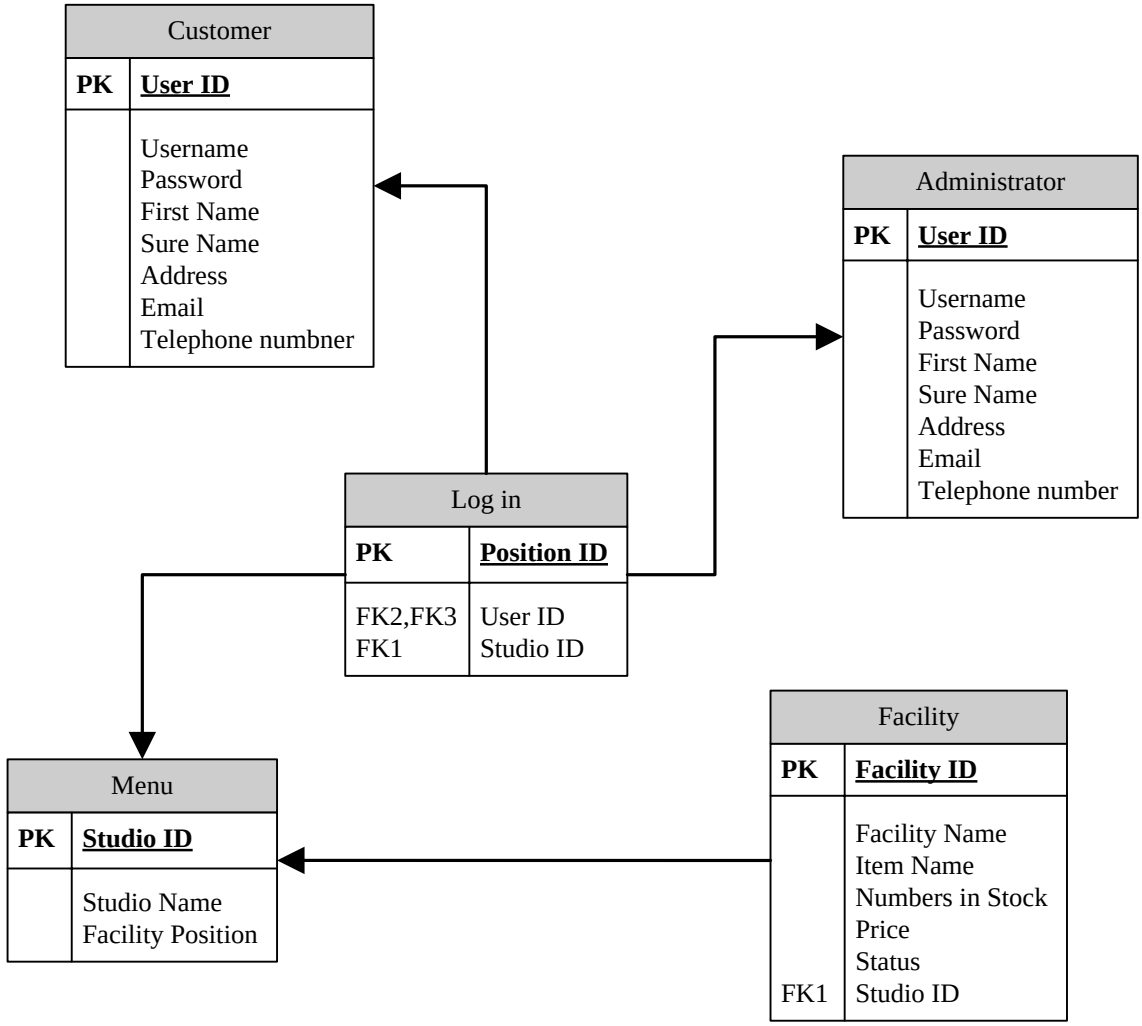

Figure 7. Database Structure

4.4. Use Cases and Interface Design

It is the application working diagrams once you log in this system. The system's interface could be divided into several parts. There is a buffer used to import and export commands. Normally, these commands are been send by mobile phone. While the sever receives the command, it will automatically make relevant actions, such as update database, delete information from database and so on.

Figure 8 is showing the main structure of relationships between each interface, and this system will goes as this map designed.

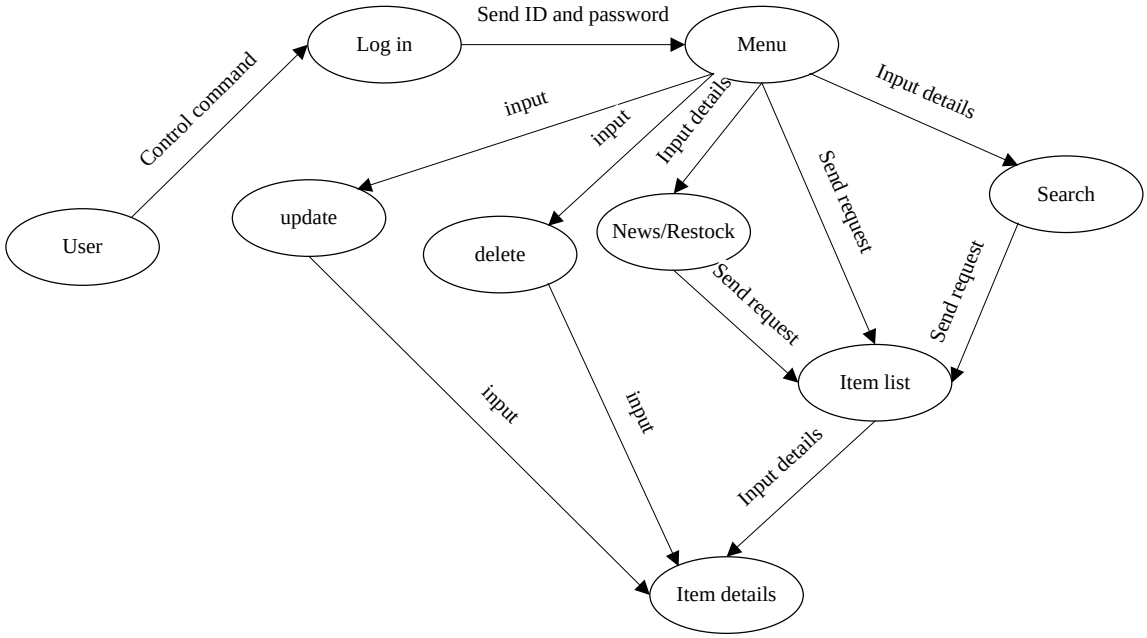

Figure 8. The Main Structure of the Relationships between Each Interface

4.4.1. Log in

As a customer or an administrator, the application must be rather secure. The common solution is either creating a log engine or invoked in command application. In general, the application is only allowed to log in by the person who has the priority to register.

Table 6 is the use case of the log in state; it will describe the expected normal process and make out a plan for unexpected things which will be happened in this state. The special requirements are also needed.

Use Case ID:	UC1		
Use Case Name:	Log In		
Created By:	Li Xu	Last Updated By:	Li Xu
Date Created:	Sptember.22.2010	Date Last Updated:	September.22.2010

Actors:	User and Administrator
Description:	The user or administrator types his/her username and password to log in the system.
Trigger:	
Preconditions:	The user or administrator types his/her username and password. Or the user and administrator select log in or delete to retype his/her password.
Postconditions:	The username and the password are correct and log in the system. Or the username or the password is not correct and he/she should retype them.
Normal Flow:	1.0 The Log in is successful 1. The user types his/her password and log in the system. 2. System accepts the requests if they are correct or refuse the request.
Alternative Flows:	1.1 The Log in is unsuccessful because the username or password is wrong 1. The system displays out an error message. 2. The user clicks 'delete' and retype his/her new password and username.
Exceptions:	1. The system is updating and not accepts any requests. 2. The username dose not exits.
Includes:	None
Priority:	High
Frequency of Use:	
Business Rules:	
Special Requirements:	The user or administrator's name and his/her password must be exit in the system's database.
Assumptions:	
Notes and Issues:	

Table 6. The Use Case of Log In State

Figure 9 is showing the application of Log In state. The programming will follow the request which are planed in the application

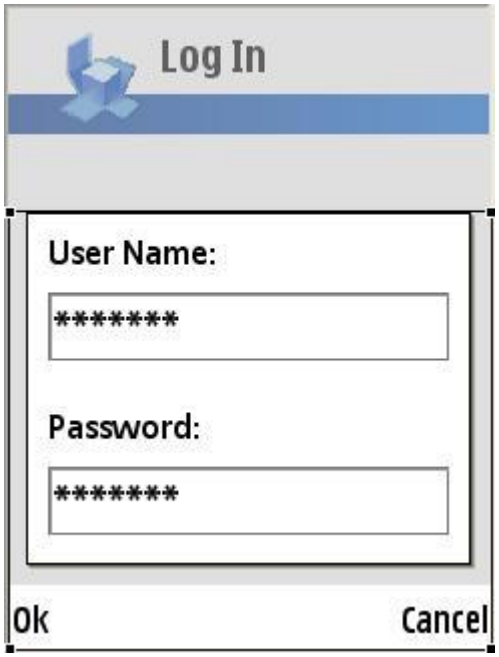

Figure 9. The Application of Log In State

4.4.2. Menu

Once you log in successful, there are two kinds of menu for each person. One is for customer and another is for administrator. It can sore out the items which the guests want to look for. And for administrator, their authority has updated, they can either update or delete the information of the items which are stored in the database.

Table 7 is the use case of Menu state; it will describe the expected running process of this state and make out a plan for unexpected things which will be happened in this state. And also make out the special requirements for different conditions.

Use Case ID:	UC2		
Use Case Name:	Menu		
Created By:	Li Xu	Last Updated By:	Li Xu
Date Created:	Sptember.22.2010	Date Last Updated:	September.22.2010

Actors:	User and Administrator
Description:	User or Administrator logged in the system and select the bar on the menu list.
Trigger:	
Preconditions:	The user or administrator selects the bar on the menu list which he/she wants to jump to. Or the user or administrator selects log out to quit the system
Postconditions:	The system accepted the user or administrator's request and jump to the interface which they selected.
Normal Flow:	2.0 The response is successful 1. The user or administrator selects the bar on the menu list and jump to the interface which he/she selected. 2. The system accept accepts the requests jump to the interface which as been selected.
Alternative Flows:	2.1 The response is unsuccessful 1. The interface stays on the menu page. 2. The system displays out an error message.
Exceptions:	If the interface which the user wants to jump to is updating, the system will not accept the user's requirement.
Includes:	None
Priority:	High
Frequency of Use:	
Business Rules:	
Special Requirements:	The Administrator has more bars the user's to administrate the system.
Assumptions:	
Notes and Issues:	

Table 7. The Use Case of Menu State

Figure 10 is the application of menu for customer; customers can choose the selection on this menu and find out the results from each bar.

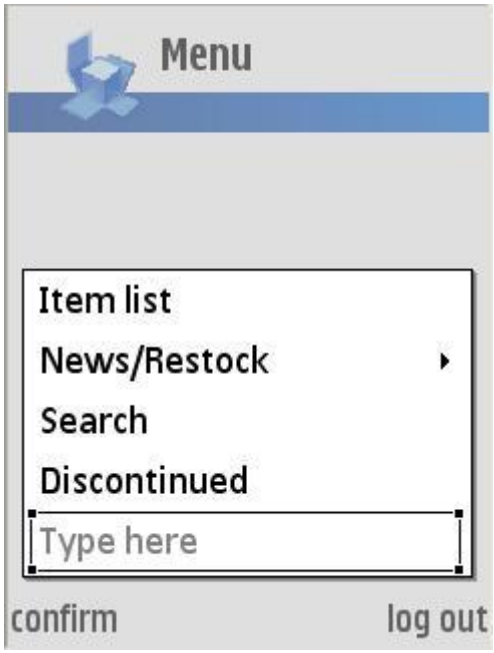

Figure 10. The Application of Menu for Customer

Figure 11 is the application of menu for administrator, compare with the menu for customer, it provide the additional choice for administrator to update or delete information directly form database by using their own smart phone.

Figure 11.The Application of Menu for Administrator

4.4.3. Item List

The item list will give out all the item which the users wants to look at, and let the users to choose what name of the items they would like to have a look at. They can either jump back to the menu or check the details of the item which they have selected, such as the full name, the numbers in stock, the retail price and so on.

The Table 8 below is the Use Case of the Item List state, the purpose of this table is the same to plan out the expected normal process and prepare for the unexpected things which will be happened in the Item List state and also create some special requirments of this state.

Use Case ID:	UC3		
Use Case Name:	Item List		
Created By:	Li Xu	Last Updated By:	Li Xu
Date Created:	Sptember.22.2010	Date Last Updated:	September.22.2010

Actors:	User
Description:	User and Administrator check the items from the list.
Trigger:	
Preconditions:	The user or administrator selects the item name which they want to check or jump back to the menu.
Postconditions:	The system accepted the user or administrator's request and jump to the interface which they selected.
Normal Flow:	3.0 The response is successful 1. The user and administrator make his/her decision which interface they want to go. 2. The screen jumps to the interface list out the detail of the item which he/she selected or jumps back to the menu interface.
Alternative Flows:	3.1 The response is unsuccessful 1. The interface stays at the item list page. 2. The system displays out an error message.
Exceptions:	If the item dose not exits in the database, then the system will not accept the user or administrator's requirement.
Includes:	None
Priority:	High
Frequency of Use:	
Business Rules:	
Special Requirements:	The item's data must be exits in the system's data base.
Assumptions:	
Notes and Issues:	

Table 8. The Use Case of Item List State

Figure 12 is showing the application of Item List state. The interface of Item List state will be the same as this application after the programming has been completed.

Figure 12. The Application of Item List State

4.4.4. Item Details

Once the user confirmed the items what they want to check, the system will map out the full name of the items, the numbers of the item in stock and the price of the items. It will be clearly to show the customers of the details of the item. But in another way, it can only show the item's detail, if the customer wants to buy it, this system cannot solve this problem.

Table 9 is the Use Case of Item Details State. The purpose of this table is the same as the previous use case tables in this chapter, plan out the expected normal process and prepare for the unexpected things which will be happened in this state.

Use Case ID:	UC4		
Use Case Name:	Item Detail		
Created By:	Li Xu	Last Updated By:	Li Xu
Date Created:	Sptember.22.2010	Date Last Updated:	September.22.2010

Actors:	User and Administrator
Description:	The user and administrator can see the item's detail which they want to check.
Trigger:	
Preconditions:	The user or administrator selects 'back'.
Postconditions:	The system accepts the request and jump to the previous interface.
Normal Flow:	4.0 The response is successful The user or administrator select 'back' and jump back to the previous interface.
Alternative Flows:	4.1 The response is unsuccessful 1. The system displays out an error message. 2. The interface will automatically jump back to the menu
Exceptions:	Some times the connection to the server is not stable, and long time has no response.
Includes:	None
Priority:	High
Frequency of Use:	
Business Rules:	
Special Requirements:	None
Assumptions:	
Notes and Issues:	

Table 9. The Use Case of Item Detail State

Figure 13 is showing the application of Item Details state. The interface of Item Details state will be the same as this application after the programming has been completed.

Figure 13. The Application of Item Detail State

4.4.5. News/Restock

Some customers would like to check the new items which are just been put on sell and look for the items which have discontinued for a long time but now back from stock. In this state, the system will list out the items which are new and which are back from stock, so that can let the customer to choose what they would like the check and show the details of the items which they have selected.

Table 10 is the Use Case of News/Restock state, plan out the expected normal process and prepare for the unexpected things which will be happened in this state, also create the special requirements for this state is the purpose of this table.

Use Case ID:	US5		
Use Case Name:	News/Restock		
Created By:	Li Xu	Last Updated By:	Li Xu
Date Created:	Sptember.22.2010	Date Last Updated:	September.22.2010

Actors:	User and Administrator
Description:	The user and administrator check the item list in this interface.
Trigger:	
Preconditions:	The user or administrator selects the item name which they want to check and clicks 'confirm' or click 'back' to back to the menu.
Postconditions:	The system accepted the user and administrator's request and jump to the interface which they selected.
Normal Flow:	5.0 The response is successful 1. The user or administrator makes his/her selection and the system accepts the requests. 2. The interface jumps to the interface list out the detail of the item which he/she selected or jump back to the menu.
Alternative Flows:	5.1 The response is unsuccessful 1. The interface stays at the News/Restock page. 2. The system displays out an error message.
Exceptions:	If the item dose not exits in the database, then the system will not accept the user or administrator's requirement.
Includes:	
Priority:	High
Frequency of Use:	
Business Rules:	
Special Requirements:	The item's data must be exits in the system's data base.
Assumptions:	
Notes and Issues:	

Table 10. The Use Case of News/Restock State

Figure 14 is showing the application of the News/Restock state. The interface of News/Restock state will be the same as this application after the programming has been completed.

Figure 14. The Application of News/Restock State

4.4.6. Search

The search state will let the user or administrator who has known the specific name of the items. So the user can find out the information of the items what that want to looking for as fast as possible.

Table 11 is the Use Case of Search state, the purpose of this table is to plan out the expected normal process and prepare for the unexpected things which will be happened in this state; also create the special requirements for this state.

Use Case ID:	SMA6		
Use Case Name:	Search		
Created By:	Li Xu	Last Updated By:	Li Xu
Date Created:	Sptember.22.2010	Date Last Updated:	September.22.2010

Actors:	User and Administrator
Description:	The user or administrator types in the item's name and searches the results.
Trigger:	
Preconditions:	The user or administrator type in the item's name and click 'search' or click 'back'
Postconditions:	The system accepts the request. Search start and list out the result or the interface jump back to menu.
Normal Flow:	6.0 The response is successful 1. The user or administrator type in the item's name and system accept the requirements or back to the menu 2. The system accept the request, list out the item which the user or administrator is looking for or the interface jump back to the menu.
Alternative Flows:	6.1 The response is unsuccessful 1. The interface stays at the searching page. 2. The system displays out an error message.
Exceptions:	If the searching engine in updating or the item which he/she typed in dose not exits in the system's database, the searching will not continue.
Includes:	None
Priority:	High
Frequency of Use:	
Business Rules:	
Special Requirements:	The item's data must be exits in the system's data base or the searching engine is updating
Assumptions:	
Notes and Issues:	

Table 11. The Use Case of Search State

Figure 15 is showing the application of Search state. The interface of Search state will be the same as this application after the programming has been completed.

Figure 15. The Application of Search State

4.4.7. Update

The Update state is just for administrator only. The can update the items' name, price and numbers of stock in real time. Add new items information or correct the information as the administrator expected.

Table 12 is the Use Case of the Update state, this table will also plan out the expected normal process and prepare for the unexpected things which will be happened in this state; also create the special requirements for the Update state.

Use Case ID:	UC7		
Use Case Name:	Update		
Created By:	Li Xu	Last Updated By:	Li Xu
Date Created:	Sptember.22.2010	Date Last Updated:	September.22.2010

Actors:	The Administrator
Description:	The administrator updates the information of an item in the system's database.
Trigger:	
Preconditions:	The administrator types a new item name and its price, numbers in stock and click 'update' or 'click' back to the previous interface.
Postconditions:	The system accept the administrator's request, update the item's information or back to menu.
Normal Flow:	7.0 Update successful 1. The administrator type an item name with its new price and numbers in stock, then click 'update' or back to the menu 2. The system accepts the request, new name, price and numbers in stock will be updated into database or the interface jump back to the menu.
Alternative Flows:	7.1 Update unsuccessful 1. The interface stays at the searching page. 2. The system displays out an error message.
Exceptions:	The connection is unstable or the database is updating.
Includes:	None
Priority:	High
Frequency of Use:	
Business Rules:	
Special Requirements:	If the item name exits in the database, it will just update the price and numbers in stock.
Assumptions:	
Notes and Issues:	

Table 12. The Use Case of Update State

Figure 16 is showing the application of Update state. The interface of Update state will be the same as this application after the programming has been completed.

Figure 16. The Application of Update State

4.4.8. Delete

This state is only for administrator, too. If there are some items out of stock, the administrator can delete the information from the database.

Table 13 is the Use Case of Delete state. It will plan out the expected normal process and prepare for the unexpected things which will be happened in this state; also create the special requirements for the Delete state.

Use Case ID:	UC8		
Use Case Name:	Delete		
Created By:	Li Xu	Last Updated By:	Li Xu
Date Created:	Sptember.22.2010	Date Last Updated:	September.22.2010

Actors:	The Administrator
Description:	The administrator deletes the information of an item in the system's database.
Trigger:	
Preconditions:	The administrator type an item name and click 'delete' or click 'back'.
Postconditions:	The system accepts the administrator's request and deletes the item's information from database or jump back to menu.
Normal Flow:	8.0 Delete successful 1. The administrator types an item name then click 'delete' or back to the menu. 2. The system accept the administrator's request and delete the item's information from data base or the interface jump back to menu.
Alternative Flows:	8.1 Delete unsuccessful 1. The interface stays at the searching page. 2. The system displays out an error message.
Exceptions:	The item name is not exits in database or the database is updating.
Includes:	
Priority:	High
Frequency of Use:	
Business Rules:	
Special Requirements:	The item name must be exited in the system's database.
Assumptions:	
Notes and Issues:	

Table 13. The Use Case of Delete State

Figure 17 is showing the application of the Delete state. The interface of Delete state will be the same as this application after the programming has been completed.

Figure 17. The Application of Delete State

4.4.9. Discontinued

The discontinued state will show the user which item are discontinued. The items in the discontinued list will not be shown in the other list. And the users also and check the details of the items in the discontinued list.

The Table 14 below is the Use Case of Discontinued state, the purpose of this table is to plan out the expected normal process and prepare for the unexpected things which will be happened in this state; also create the special requirements for the Discontinued state.

Use Case ID:	UC9		
Use Case Name:	Discontinued		
Created By:	Li Xu	Last Updated By:	Li Xu
Date Created:	Sptember.22.2010	Date Last Updated:	September.22.2010

Actors:	User and Administrator
Description:	The user and administrator check the item list in this interface.
Trigger:	
Preconditions:	The user or administrator selects the item name which they want to check and clicks 'confirm' or click 'back' to back to the menu.
Postconditions:	The system accepted the user and administrator's request and jump to the interface which they selected.
Normal Flow:	9.0 The response is successful 1. The user or administrator makes his/her selection and the system accepts the requests. 2. The interface jumps to the interface list out the detail of the item which he/she selected or jump back to the menu.
Alternative Flows:	9.1 The response is unsuccessful 1. The interface stays at the Discontinued page. 2. The system displays out an error message.
Exceptions:	If the item dose not exits in the database, then the system will not accept the user or administrator's requirement.
Includes:	
Priority:	High
Frequency of Use:	
Business Rules:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Table 14. The Use Case of Discontinued State

Figure 18 is showing the application of Discontinued state. The interface of Discontinued state will be the same as this application after the programming has been completed.

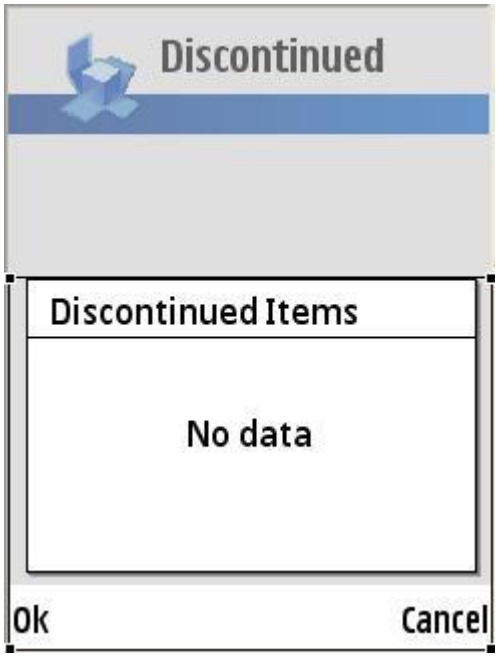

Figure 18. The Application of Discontinued State

4.5. Others

The foundational application controllers are keyboards of mobile phone in the implementation of these modules. The smart phone always includes the common keyboards of the inputs as the normal phone.

The creation of database has not been completed yet, so there are no items can be found in the list.

Server has not been built. Although the project processed for a long time, only the Design of Mobile Application has been done. So there is no way to test the connection between the system and Server.

If this kind of devices has serious problems, the principal must be around the devices and fix it. It is a limitation of mobile software.

5. TESTING

The mobile software installation must be accomplished by the compiler platform, which this project was processed by Carbide C++ & Qt.

A test plan is a system approach to testing a system such as this one. The plan typically contains a detailed understanding of what the eventual workflow will be. I this project, the testing plan have two parts; they are black box testing and white box testing.

5.1. Black Box Testing

The task of black box testing in this project is about to test the internal design or code. All the interfaces need to be tested when they are been completed one by one. We have to make sure the code can run smoothly. After that we have to check the demo on the Carbide C++ and Qt Emulator, which will run the software on a simulated smart phone key board. The process is described in Figure 19.

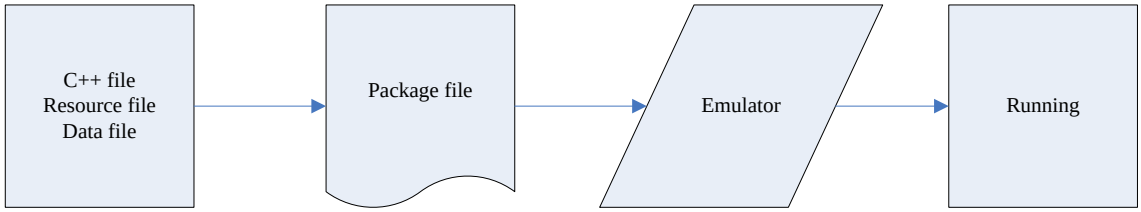

Figure 19. The Process of Black Box Testing

The resource file is a standard and convenient way to distribute Symbian applications, users can install/uninstall a recourse package either from a PC using connectivity software, or from a Symbian OS phone using an on-board installation program. The resource file has embedded mechanism that allows deployment of shared libraries or application upgrades.

The package file is the key to create a correct package. It can be generated automatically using the Resource Automatic Running tool or written manually. The Nokia Application Wizard creates a default PKG file when you use it to start a project.

The testing cases will make a clear clue about the details to test this system. It will set test inputs, execution conditions, and expected results developed for a particular objective: to exercise a particular program path or verify compliance with a specific requirement.

The Table 15 below is the Test Cases of this whole system, the Test Case ID number will be matched with the Use Case number in chapter 4.4. This test case will describe the sequence of action which is performed by this Super Marketing Administration System. It will provide an absolute result which the unexpected things happened, and the condition of each user and for each scenario.

Test Case ID	Scenario/ condition	User condition	Prerequisites fulfilled	Expect Result
TC1	Invalid user name and password	User and Administrator	Yes	Stay at this step, display out an error message.
TC 2	Succeed log in	User and Administrator	Yes	Interface jump into the one which the user and administrator want to.
TC 3	The item list has been selected in menu	User and Administrator	Yes	The table will sore out the item name in the item list.
TC 4	The item's detail has been sore out	User and Administrator	N/A	The item's detail has been sore out.
TC 5	The item is not exit in the database	User and Administrator	N/A	The item's detail will not be sore out. Stay at this step, display out an error message..
TC 6	The system is updating	User and Administrator	N/A	Stay at this step, the system will not accept any requirement from any user, and display out an error message.
TC 7	The request not be accepted	Administrator	N/A	Stay at this step, display out an error message.
TC 8	The request conflict	Administrator	N/A	Stay at this step, display out an error message.
TC 9	The prerequisites fulfilled	User and Administrator	N/A	Stay at this step, display out an error message.

Table 15. The Test Cases

5.2. White Box Testing

The following steps will show weather this project is successful or not, it is also one of the main objectives in this project. Just like the previous chapters have been mentioned, the key part of the project is the short messages can be send out between each interfaces successfully or not.

- Start the emulator; it should open the log in interface autonomously.
- Log in and go to the main menu.
- Select each icon on the list to check if it can get the item's detail from the database.
- Quit the emulator when the interface can not jump to another.

- Modify the code and the resource file.
- Reprogram the code of the interface with the modified resource file.
- The data must make sure user can get the correct information of the items which they selected in the menu from the database.

6. CONCLUSIONS

The objective of this project is to use Qt and Carbide C++ to design the interfaces of mobile software which is implemented as an administration system. It also showed both weakness and strength of this supermarket administration system.

Compare with Carbide C++, Qt is totally new software for me. Because I am not familiar with it, so I only use it when I complete the application design in Carbide C++, I will use it for the future coding plan, and also use it as an emulator, although Carbide C++ is also en emulator. But learning how to use Qt is importing for the future programming.

There are some problems occurred in the process:
In the designing part, each interface has its own application. The focus is that each icon in the application has it own code in the future coding, because they need to run with their own codes and a database.

The time of this project seems not enough for me and I just have done the basic application design. The system can add some more use abilities to improve it. The implementation of the programming will be the future of development of this project. The task for the future development is to add their codes and database on to make it as a real system.

In the testing part, the testing process will go through as which have been planned in chapter 5. So in the testing part, the designer will follow the process of the testing plan after the programming has been completed and they could get a clear clue about the status of the system, what is going on, and so on. During the testing process, the designer will modify the codes and resource file until the codes are running as expected.

7. REFERENCES

List of the sources:

/1/ GPRS [WWW-document]
< http://www.telecomspace.com/datatech-gprs.html > 22/9/2010

/2/ Clint Smith, Daniel Collins, 3G Wireless Networks --- The McGraw Companies, 2002

/3/ GSM Technical Data [WWW-document]
< http://www.cellular.co.za/gsmtechdata.htm> 22/9/2010

/4/ Symbian OS [WWW-document] < http://www.allaboutsymbian.com/> 22/9/2010

/5/ Nokia forum Symbian OS Communication Architecture [WWW-document]
<http://wiki.forum.nokia.com/index.php/Symbian_OS_Communication_Architecture>
22/6/2010

/6/ Nokia forum Types of Memory in Symbian OS [WWW-document]
<http://wiki.forum.nokia.com/index.php/Types_of_Memory_in_Symbian_OS>
27/6/2010

/7/ Harrison, Richard, Symbian OS C++ for Mobile Phones --- Volume2, 1st Edition, John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, 2004

/8/ Nokia forum Tool Details [WWW-document]
<http://www.forum.nokia.com/info/sw.nokia.com/id/dbb8841d-832c-43a6-be13-f78119a2b4cb.html> 1/10/2010

/9/ Qt Nokia Products [WWW-document]
<http://qt.nokia.com/products> 1/10/2010

/10/ Nokia forum Symbian S60 SDK [WWW- document]
<http://www.forum.nokia.com/Library/Tools_and_downloads/Other/Symbian_SDKs/
> 1/10/2010

List of the figures:

8. LIST OF APPENDICEX

APPENDIX 1 **Data Definition of Database**

APPENDIX 1

Data Definition of Administrator

The definitions of data are related with the table 'Administrator' in Figure 7.

Name	Primary Key	Foreign Key	Type	Not Null
User ID	Yes	No	Char(10)	Yes
Username	No	No	Char(10)	Yes
Password	No	No	Char(10)	Yes
First Name	No	No	Char(10)	Yes
Sure Name	No	No	Char(15)	Yes
Address	No	No	Char(35)	Yes
Email	No	No	Char(35)	Yes
Telephone number	No	No	Char(10)	Yes

Data Definition of Customer

The definitions of data are related with the table 'Customer' in Figure 7.

Name	Primary Key	Foreign Key	Type	Not Null
User ID	Yes	No	Char(10)	Yes
Username	No	No	Char(10)	Yes
Password	No	No	Char(10)	Yes
First Name	No	No	Char(10)	Yes
Sure Name	No	No	Char(15)	Yes
Address	No	No	Char(35)	Yes
Email	No	No	Char(35)	Yes
Telephone number	No	No	Char(10)	Yes

Data Definition of Log In

The definitions of data are related with the table 'Log In' in Figure 7.

Name	Primary Key	Foreign Key	Type	Not Null
Position ID	Yes	No	Char(10)	Yes
User ID	No	No	Char(10)	Yes
Studio ID	No	No	Char(10)	Yes

Data Definition of Menu

The definitions of data are related with the table 'Menu' in Figure 7.

Name	Primary Key	Foreign Key	Type	Not Null
Studio ID	Yes	No	Char(10)	Yes
Studio Name	No	No	Char(20)	Yes
Facility Position	No	No	Char(20)	Yes

Data Definition of Facility

The definitions of data are related with the table 'Facility' in Figure 7.

Name	Primary Key	Foreign Key	Type	Not Null
Facility ID	Yes	No	Integer(5)	Yes
Facility Name	No	No	Char(10)	Yes
Item Name	No	No	Char(20)	Yes
Numbers in Stock	No	No	Integer(5)	Yes
Price	No	No	Integer(5)	Yes
Status	No	No	Char(10)	Yes
Studio ID	Yes	No	Char(10)	Yes